UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/606,961	06/29/2000	James P. Rodrigues	MS 150530.1/40062.69US01	7182

7590 04/02/2004

Merchant & Gould PC
P O Box 2903
Minneapolis, MN 55402-0903

EXAMINER	
KISS, ERIC B	

ART UNIT	PAPER NUMBER
2122	13

DATE MAILED: 04/02/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	09/606,961	RODRIGUES ET AL.
	Examiner	Art Unit
	Eric B. Kiss	2122

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *15 January 2004*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-27* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-27* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *15 January 2004* is/are: a)☐ accepted or b)☒ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. The reply filed January 15, 2004, has been received and entered. Claims 1-27 are pending.

Response to Amendment

2. Applicant's submission of Replacement Sheets appropriately addresses the objections to the drawings as detailed in the previous Office action. Accordingly, these objections are withdrawn. However, additional objections are presented below in view of changes made by Applicant.

3. Applicant's amendments to the specification appropriately address the objections to the specification as detailed in the previous Office action. Accordingly, these objections are withdrawn in view of Applicant's amendments.

4. Applicant's amendments to the claims appropriately address the objections to the claims as detailed in the previous Office action. Accordingly, these objections are withdrawn in view of Applicant's amendments.

Drawings

5. The drawings are objected to as failing to comply with 37 CFR 1.84(p)(5) because they include the following reference sign(s) not mentioned in the description: "213" in Fig. 2. A proposed drawing correction, corrected drawings, or amendment to the specification to add the reference sign(s) in the description, are required in reply to the Office action to avoid abandonment of the application. The objection to the drawings will not be held in abeyance.

Specification

6. The disclosure is objected to because of the following informalities: reference character "212" has been used to designate both a hard disk drive and a BIOS.

Appropriate correction is required.

Response to Arguments

7. Applicant's arguments with respect to claims 1-27 have been considered but are moot in view of the new ground(s) of rejection.

Claim Rejections - 35 USC § 103

8. The text of those sections of Title 35, U.S. Code not included in this action can be found in a prior Office action.

9. Claims 1-27 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,158,049 to Goodwin et al. in view of U.S. Patent No. 6,349,406 to Levine et al. and further in view of U.S. Patent No. 6,205,545 to Shah et al.

As per claim 1, *Goodwin et al.* disclose a computing system (see Fig. 1) for obtaining run-time internal state data within an application program, the computing system comprising:

a init module for determining if the run-time internal state data is to be collected during the operation of the application program (see registry entry description in column 10, line 59 through column 11, line 42);

a performance code marker module for obtaining and storing the run-time internal state data for later retrieval (see column 6, lines 46-64); and

an uninit module for formatting and storing the obtained run-time internal state data into memory that permits retrieval after the termination of the application program (profile data is stored in a profile optimizer database; see column 6, lines 50-52);

wherein

the init module is executed before any run-time internal state data is collected (the application program is instrumented before it is executed); and

the performance code marker module is executed each time run-time internal state data is to be collected (profile data is generated during execution; see Fig. 2).

Goodwin et al. fail to expressly disclose the uninit module being executed after all run-time internal state data desired has been collected. However, *Levine et al.* teach formatting and storing obtained run-time internal state data after tracing is finished (sending buffer contents to a file and generating a report; see Figs. 4 and 6 and the associated text in columns 9 and 11). Therefore, it would have been obvious to one having ordinary skill in the computer art at the time the invention was made to modify the system of *Goodwin et al.* to include formatting and storing obtained run-time internal state data after tracing is finished as per the teaching of *Levine et al.* One would be motivated to do so to reduce computational overhead while executing a debugger process.

Goodwin et al. fail to expressly disclose the predefined points corresponding to permanently inserted performance markers. However, *Shah et al.* teach a runtime optimization strategy that allows an instrumented program to run in an optimized manner, with debugging toggled via a debug flag, such that the additional debugging code stays in the native code pool where it is rarely executed (see, for example, col. 8, lines 53-65). In this manner, the code remains instrumented, but the debugging features that rely on the instrumentation can be disabled through the debug flag, reducing the overhead required. Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to further modify the system of *Goodwin et al.* to include permanently inserted performance markers as per the teachings of *Shah et al.* One would be motivated to do so to gain the advantages of retaining debugging features while reducing overhead associated with those features.

As per claim 2, *Goodwin et al.* further disclose the init module determining if run-time internal state data is to be collected (see registry entry description in column 10, line 59 through column 11, line 42). Therefore, for reasons stated above, such a claim also would have been obvious.

As per claims 3 and 4, *Goodwin et al.* further disclose the init module making the determination that run-time internal state data is to be collected by checking for the existence of an identification key within a system registry and checking for the existence of processing modules identified by the identification key (the registry key points to the instrumented executable and instructs the operating system to run the instrumented version; see column 11, lines 16-42; if the instrumented module identified by the system registry does not exist, it is inherent that tracing will not proceed). Therefore, for reasons stated above, such claims also would have been obvious.

As per claim 5, *Goodwin et al.* further disclose the performance code marker module collecting run-time internal state data only if the init module has determined that the run-time internal state data is to be collected (if the registry key instructing the operating system to run the instrumented executable is not present, the operating system executes the non-instrumented version; see column 10, line 59 through column 11, line 42). Therefore, for reasons stated above, such claims also would have been obvious.

As per claims 6-8, in addition to the disclosure and teachings applied above, *Goodwin et al.* further disclose generating, storing, and retrieving a performance data record containing the collected-run time internal state data (profile data is stored in a profile optimizer database as

records; see column 6, lines 50-52). Therefore, for reasons stated above, such claims also would have been obvious.

As per claims 9, 10, and 12, *Goodwin et al.* fail to expressly disclose the run-time internal state data comprising benchmark timing data, memory usage data, and open file usage data. However, *Levine et al.* further teach the use of a trace tool to gather such data (see column 15, lines 1-10; and column 17, lines 26-35; benchmark timing data and memory usage are furthermore considered to be related to the state of the currently open files). Therefore, it would have been obvious to one having ordinary skill in the computer art at the time the invention was made to further modify the system of *Goodwin et al.* to include gathering and processing benchmark timing data and memory usage data as per the teachings of *Levine et al.* One would be motivated to do so to be able to determine how and when system resources are being used.

As per claim 11, *Goodwin et al.* disclose run-time internal state data comprising system registry usage data (see column 10, lines 63-67). Therefore, for reasons stated above, such claims also would have been obvious.

As per claim 13, *Goodwin et al.* disclose a method for obtaining run-time internal state data within an application program, the method comprising:

inserting one or more code markers into the application program at locations within the application program corresponding to the point at which run-time internal state data is desired (instrumenting the code; see column 10, lines 40-51);

determining if run-time internal state data is to be collected at each code marker by checking for the existence of processing modules identified by an identification key within a system registry (see registry entry description in column 10, line 59 through column 11, line 42);

if the run-time internal state data is to be collected at each code marker:

generating a performance data record containing the collected run-time internal state data each time the code markers are reached (see column 6, lines 46-64);

Goodwin et al. fail to expressly disclose storing the performance data records within a data memory block within the processing modules and retrieving the performance data records from the data memory block for transfer to a mass storage device once all of the run-time internal state data has been collected. However, *Levine et al.* teach the use of a trace data buffer allocated by the trace processor for storing trace data generated during a debugging process and outputting the date from the buffer to a file for post-processing after tracing is complete (see Fig. 6 and its associated text in column 11). Therefore, it would have been obvious to one having ordinary skill in the computer art at the time the invention was made to modify the method of *Goodwin et al.* to include storing performance data records in a data memory block within a processing module for subsequent transfer to a mass storage device upon completion of tracing as per the teachings of *Levine et al.* One would be motivated to do so to reduce computational overhead while executing a debugger process.

Goodwin et al. fail to expressly disclose the predefined points corresponding to permanently inserted performance markers. However, *Shah et al.* teach a runtime optimization strategy that allows an instrumented program to run in an optimized manner, with debugging toggled via a debug flag, such that the additional debugging code stays in the native code pool where it is rarely executed (see, for example, col. 8, lines 53-65). In this manner, the code remains instrumented, but the debugging features that rely on the instrumentation can be disabled through the debug flag, reducing the overhead required. Therefore, it would have been obvious

to one of ordinary skill in the art at the time the invention was made to further modify the system of *Goodwin et al.* to include permanently inserted performance markers as per the teachings of *Shah et al.* One would be motivated to do so to gain the advantages of retaining debugging features while reducing overhead associated with those features.

As per claims 14-17, see the rationale applied above with respect to claims 9-12.

As per claim 18, this is a product version of the claimed method discussed above (claim 13). Furthermore, such a computer-readable product is inherently required by the system of *Goodwin et al.*, and all other limitations have been addressed as set forth above. Therefore, for reasons stated above with respect to claim 13, such a claim also would have been obvious.

As per claims 19 and 20, see the rationale applied above with respect to claims 3 and 4.

As per claim 21, *Goodwin et al.* fail to expressly disclose the data memory block being within the processing module. However, *Levine et al.* teach the use of a trace data buffer allocated by the trace processor for storing trace data generated during a debugging process. Therefore, it would have been obvious to one having ordinary skill in the computer art at the time the invention was made to further modify the product of *Goodwin et al.* to include the data memory block being within the processing module as per the teaching of *Levine et al.* One would be motivated to do so to reduce computational overhead while executing a debugger process.

As per claims 22-25, see the rationale applied above with respect to claims 9-12.

As per claim 26 and 27, official notice is taken that it was well known and practiced at the time the invention was made to encode computer program instructions on such computer-readable storage media and propagated signals on carriers for the purpose of storing and

transmitting the instructions during their implementation. Therefore, it would have been obvious to one having ordinary skill in the computer art at the time the invention was made to further modify the product of *Goodwin et al.* to include such storage media and propagated signals as they are well-suited to embodying such program instructions.

Conclusion

10. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

11. Any inquiry concerning this communication or earlier communications from the Examiner should be directed to Eric B. Kiss whose telephone number is (703) 305-7737. The

Examiner can normally be reached on Tue. - Fri., 7:30 am - 5:00 pm. The Examiner can also be reached on alternate Mondays.

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, Tuan Dam, can be reached on (703) 305-4552. The fax phone number for the organization where this application or proceeding is assigned is (703) 872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

EBK/EBK
March 30, 2004

WEI Y. ZHEN
PRIMARY PATENT EXAMINER